

Rubens de Melo Galani

DESENVOLVEDOR FULL STACK

+55 11 93494-2642

Cotia, SP

Desenvolvendo aplicações web escaláveis desde 2023 com foco em negócios e projetos criativos.

 rubensgalani@gmail.com

 github.com/JJRgb0

 rubens-dmg.vercel.app

TECNOLOGIAS

- React
- Javascript
- Typescript
- Sass
- Tailwind CSS
- NextJS
- NodeJS
- ExpressJS
- Prisma
- SQL
- Docker
- Git/Github

EXPERIÊNCIA

kdLux

Desenvolvedor Full Stack • Out 2023-Abr 2024

- Desenvolvi sozinho uma plataforma e-commerce full stack (Next.js 13 App Router) com autenticação (Google OAuth + credenciais próprias) usando NextAuth e Prisma sobre PostgreSQL.
- Modelei e implementei um schema relacional complexo (13+ modelos: Users, Products, Variants, Options, Collections, Purchases, Payments, Coupons, Comments, Threads etc.) garantindo relações e integridade via Prisma.
- Implementei fluxo de checkout multi-etapas (endereço, carrinho/cupom, pagamento) com cálculo dinâmico de frete (API MelhorEnvio) e aplicação de cupons percentuais ou valor fixo.
- Integrei pagamentos Mercado Pago (cartão, PIX, boleto) incluindo criação, consulta e cancelamento de transações via rotas serverless e tokens seguros.
- Estruturei autenticação com sessões JWT, adapter Prisma e hashing de senha (bcrypt), incluindo controle de rotas protegidas e fallback de login.
- Otimizei performance usando rotas Edge Runtime (produto, busca, home, OG image) e carregamento incremental (Suspense / streaming).
- Criei sistema de gerenciamento de catálogo com variantes, estoque, especificações e recomendações de produtos, além de filtro por múltiplas categorias (estilo, tipo, gênero).
- Desenvolvi painéis administrativos com upload de imagens via Cloudinary (form-data + otimização de assets) e manipulação de coleções/produtos.
- Implementei carrinho persistido no cliente (localStorage) com lógica de agregação inteligente de itens e variantes.
- Apliquei SEO técnico (metadata dinâmica, JSON-LD para produtos, sitemap, robots, OpenGraph).
- Integrei envio de emails transacionais via SMTP seguro (Zoho + Nodemailer) e fluxo de verificação / recuperação.

- Padronizei código com ESLint (incluindo plugin unicorn) + Prettier + lint-staged em pre-commit para garantir qualidade contínua.
- Construí camada de APIs internas tipadas (fetch server-side com headers de autorização) segregando domínio (lib/kdLux) e adaptando para deploy em Vercel.
- Assegurei escalabilidade e depuração adicionando logging de queries Prisma e separação de variáveis seguras (server actions para acesso a secrets).
- Implementei lógica de recomendação e listagem paginada de coleções / produtos com ordenação dinâmica (sortKey + reverse) e filtros combináveis.
- Adicionei suporte a avaliações e threads de comentários por produto, incluindo cálculo de notas e relação usuário–pedido.
- Utilize Tailwind CSS (com tailwind-merge e variants) para construir UI responsiva e reutilizável com consistência visual.

GoldenAce

Líder (projeto pessoal) • Abr 2025-Jun 2025

- Desenvolvi sozinho uma plataforma full stack de arcade 3D interativo (React + Three.js) com integração de jogo Unity WebGL e sistema de leaderboard seguro (Node.js + Prisma + PostgreSQL).
- Implementei API REST em TypeScript (Express 5) com validação de integridade via hash SHA-256 para submissão de pontuações e lógica eficiente de Top 10 (inserção seletiva e pruning).
- Modelei persistência com Prisma (migrations versionadas) e utilizei variáveis de ambiente para configurar ambiente (dev/prod) com build separado (Vite + scripts TS).
- Integrei jogo Unity dentro da UI React via iframe com canal de mensagens e injeção programática de eventos de teclado (bridge custom) sincronizando binds dinâmicos do usuário.
- Criei sistema de mapeamento e reconfiguração de controles (Redux Toolkit + localStorage) permitindo persistência, hot reload de binds e propagação a Unity via mensagens estruturadas.
- Construí interface 3D imersiva (React Three Fiber + Drei) com gerenciamento de estado global e components desacoplados para cena, HUD e navegação pseudo-arcade.
- Apliquei boas práticas: modularização (controllers / services / routes), separação de camadas, hashing de dados sensíveis, safe defaults e limpeza de event listeners.

FORMAÇÃO

Formação independente em desenvolvimento web, com foco em projetos práticos e estudo contínuo de tecnologias modernas.

[Harvard University – CS50: Introduction to Computer Science](#)

IDIOMAS

 **Português**

Nativo

 **Inglês**

Intermediário/Avançado